(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(54) **Title:** METHOD AND APPARATUS FOR DETERMINING THE CORRECT OPERATING SOFTWARE VERSION FOR A NETWORK DEVICE

(57) **Abstract:** An apparatus and method for determining whether the software (122) to be downloaded by a device is the correct software (122) for that device. The software (122) includes a header portion further including fields identifying the device type (112), software type (116), and device class (120) to which the software pertains. The downloading device checks both the device (112, 120) and software type (116) before processing with the download process. If the device type (112) and software type (116) are correct, the device then checks the hardware class (120) of the software. If this last comparison process identifies a match, then the software is downloaded (120) to the device. Following the download process an error detection (124) (and optionally error correction) is performed on the software using a checksum in the header.

patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— with international search report

— before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

1

METHOD AND APPARATUS FOR DETERMINING THE CORRECT
OPERATING SOFTWARE VERSION FOR A NETWORK DEVICE

5   BACKGROUND OF THE INVENTION

The present invention is related generally to cable modems, and more specifically to a method and apparatus for ensuring that the cable modem has downloaded the correct operating software version.

Internet access via a telephone modem is available today at speeds up to 56
10   Kbps. The telephone-based modem modulates and demodulates data signals for transmission over the voice-band based telephony network. By contrast, a cable modem provides Internet access via the cable television system, which has a higher bandwidth and therefore can operate at higher data rates than the telephone system. The cable modem provides connectivity between a user's computer and the cable
15   system headend, at which point the cable operator provides access to the Internet, via, for example a T1 transmission line. In a cable network, data transmitted from the network headend to the computer user is referred to as downstream data; data transmitted from the user to the network headend is referred to as upstream data.

Figure 1 is a block diagram of a cable system 10 including a cable modem 20,
20   certain features of which are well known in the prior art. The cable system 10 includes a headend, not shown in Figure 1, from which the cable television program signals originate and that further provides a connection to the Internet or other external network. At the subscriber's premises, a splitter 14 splits the incoming signal. The first output terminal provides the television program signal for display on
25   a television 18 under control of a set top box 16.

The second output terminal from the splitter 14 provides connectivity to the cable modem 20. Downstream signals from the headend are provided to an RF (radio frequency) tuner 22, which is tuned to a frequency allocated to the cable modem 20 during the modem's start-up phase. Typically, the downstream signal uses QAM
30   (quadrature amplitude modulation) and therefore is demodulated in a QAM demodulator 24. The demodulated signal is input into a media access controller 26.

The signal from the media access controller 26 is input to a data and control logic unit 28 that controls overall operation of the cable modem 20 and further provides data control functions.

A computer 30 (or other communications device) is connected to the data and control logic unit 28 for receiving data sent in the downstream direction from the Internet, World Wide Web or other external network, and for transmitting data in the upstream direction. The outgoing data passes from the computer 30 through the data and control logic unit 28, through the media access controller 26, and finally is modulated by a QPSK (quadrature phase-shift keyed) or QAM modulator. The choice of QPSK or QAM is set forth in the configuration information provided to the cable 20. The upstream data then passes through the splitter 14 for transmission to the headend of the cable system and eventually for transmission to the Internet, World Wide Web or other external network.

In one embodiment, the downstream signal employs a 64/256 QAM signal capable of delivering up to 30 to 40 Mbps of data over a 6 MHz cable channel. Upstream data uses either QPSK or 16 QAM signaling with data rates available from 320 Kbps to 10 Mbps. Both the upstream and downstream data rates may be flexibly configured to match the data rate needs of the user. For instance, the cable modem 20 utilized by a business user could be programmed to receive and transmit at higher data rates. A residential user, on the other hand, may have a cable modem 20 configured with wider bandwidth access (and therefore a higher data rate) in the downstream direction to provide access to the external network, while limited to a lower speed for upstream data transmissions.

At a cable system headend 45, illustrated in Figure 2, upstream data from the subscribers 50 transmitted over a cable network 51 is demodulated and processed by a cable modem termination system 52 for performing data switching as it routes data from the many subscribers 50 to the Internet, World Wide Web or other external network, as shown. Similarly, the cable modem termination system 52 receives data from the external network and provides the necessary data switching of the received downstream data to the appropriate subscriber 50 via a headend transmitter 54. The headend transmitter 54 also receives the program signals (via satellite downlink, terrestrial microwave or land lines) for broadcast to all the subscribers 50. The user

3

data is carried over a 6 MHz channel (in Europe, the bandwidth standard is 8 MHz), which is the spectrum size allocated to a cable television channel for broadcast of program signals to all subscribers 50. At the subscriber's location, the program signal is received by the set top box 10, while the downstream data is separately received by the cable modem 20. See Figure 1. The RF cable modem tuner 22 tunes out the program signal from the cable modem 20 and the set top box 16 rejects the data signal. The number of upstream and downstream data channels in a given cable modem system is engineered based on the service area, the number of users, the data rate allocated to each user, and the available spectrum.

An element management system (not shown) is yet another component of the cable system 10 for configuring and managing a plurality of cable modem termination systems 52. The element management system is located at the cable headend 45. The operation of the element management system includes provisioning, day to day administration of the system, monitoring, activating alarms, and testing of the various components of the cable modem termination system 52. Generally, a single element management system is located at the cable system network operations center and can support many cable termination systems 52 in a wide geographic area.

When the cable modem 20 is powered-up, a connection is created to the cable modem termination system 52 via the cable network 51. This connection is made using the Internet protocol (IP) so that IP-formatted data from the external network, as received by the cable termination system 52, can be forwarded downstream to the cable modem 20 of a subscriber 50 via the cable network 51. After power-up, the cable modem 20 contacts a dynamic host configuration protocol server (DHCP) using the dynamic host configuration protocol. Many such DHCP servers are available on the network and the cable modem 20 simply broadcasts to all DHCP servers. Any DHCP server can answer the broadcast request. From the DHCP server, the cable modem 20 obtains an IP address, other IP related operational parameters and the address of the modem configuration file. A configuration file address points to a trivial file transfer protocol (TFTP) server from which the cable modem 20 downloads the configuration file. This file includes various modem configuration settings, such as access control information, downstream and upstream channel assignments and security configuration information.   Also, information in the configuration file

4

allows the cable modem 20 to identify its operating software (based on the cable modem vendor, model number, or other designator) and the location from which the operating software can be downloaded.

Disadvantageously, the current cable modem standard (referred to as DOCSIS, Data Over Cable Service Interface Specification) does not define an adequate process for validating the cable modem operating software, either before or after it is downloaded. Instead, the DOCSIS protocol specifies only that the download should be executed after the cable modem has gone into an encryption mode so that the download process is secure. The encryption feature can be turned on or off, as desired by the cable system operator, using an identified field in the configuration file. Whether the encryption feature is on or off, unusable or corrupted operating software can reside on the TFTP server from which the operating software is downloaded. The cable modem will not be aware that the software is unusable until the software is downloaded and the cable modem booted up. Only then will the cable modem recognize that the operating software is not usable. If the operating software is not usable, the cable modem continues to use the operating software version previously stored in memory. Certain memory devices for storing the operating software in the cable modem have a limited number of lifetime read/write cycles. Downloading an unusable version of the operating software wastes one of those lifetime read/write cycles. Finally, use of encryption techniques does not prevent a hacker from changing the operating software binary file that resides on the TFTP server, nor does it protect against file corruption from noise sources or equipment malfunctions. Instead, the encryption process merely protects against spoofing the TFTP server to a different site where unusable operating software could be located.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention can be more easily understood and the further advantages and uses thereof more readily apparent, when considered in view of the description of the preferred embodiments and the following figures in which:

Figure 1 is a block diagram of the prior art components of a cable television subscriber's site;

Figure 2 is a block diagram of a prior art cable system;

5

Figure 3 illustrates the file header associated with the present invention;

Figure 4 illustrates the fields comprising the file header of Figure 3; and

Figure 5 is a flow chart depicting the process in accordance with the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Before describing in detail the particular method and apparatus for determining the validity of the downloaded cable modem operating software in accordance with the present invention, it should be observed that the present invention resides primarily in a combination of steps and apparatus. Accordingly, the hardware components and the method steps have been represented by conventional elements in the drawings, showing only those specific details that are pertinent to the present invention so as not to obscure the disclosure with structural details that will be readily apparent to those skilled in the art having the benefit of the description herein.

The DOCSIS standard does not define the operating software file format. Therefore there remains a need for a method and apparatus to establish the file format and utilize information in the file to determine the applicability and validity of the operating software file at the earliest possible time during the file download process. According to the teachings of the present invention, an operating software file header includes operating software identification information, target hardware identification information and a checksum. Evaluation of the header information prior to downloading the operating software file and evaluation of the checksum after the download is complete, allows early determination of the applicability and the validity of the operating software file. In this way, the cable modem can avoid the lengthy download process of the prior art, which requires downloading the entire operating software file, if the file is corrupted, was tampered with, or is not the correct operating software for the cable modem.

According to the teachings of the present invention, certain evaluations are carried out on the header or first block (74 bytes in one embodiment) of the operating software file. If the evaluations determine that the file is not valid for the cable modem, the modem will immediately abort the TFTP download session and signal an error condition to the cable modem termination system 52. In the prior art, the cable

6

modem must download the entire operating software file before it can recognize that the operating software is not valid or is corrupted. Further, according to the present invention, the header includes a checksum for validating the entire operating software file once the download is complete. If the checksum analysis indicates that the file is not corrupted, the cable modem 20 immediately reboots and begins using the downloaded operating software file. In the event the checksum validation process reveals uncorrectable errors in the file, then the cable modem 20 signals an error condition to the cable modem termination system 52. Whenever an error condition is detected either by analysis of the header or the checksum, the cable modem 20 simply uses the last version of the operating software, i.e., the same operating software used during its last operating session.

Figure 3 illustrates the fields of an operating software file 68. The file 68 begins with a header field 70 that will be discussed further hereinbelow. A decompression code field 72 sets forth parameters for decompressing the operating software code once the download is complete. In one embodiment of the present invention, neither the header field 70 nor the decompression code field 72 are in a compressed format. The operating software code length is set forth in a length field 74. The checksum for the operating software code is set forth in a checksum field 76. Finally, the operating software code is in a field 78. In one embodiment, the length field 74, the checksum field 76, and the code field 78 are compressed prior to transmission.

Figure 4 shows the various components of the header field 70. Using the header field 70 in accordance with the teachings of the present invention, the operating software file can be validated upon receipt and analysis of the certain fields within header 70. A length field 82 sets forth the length of the operating software file 68, including all the components of the file as illustrated in Figure 3. A checksum field 84 provides the checksum value for the operating software file 68. A target identification field 86 identifies the device type to which the operating software applies. Exemplary device types include a cable modem and a set top box. If the target identification field 86 indicates that the software is intended for a device other than a cable modem, the download is immediately aborted, and an error signal generated as discussed above.

7

The software identifier field 88 identifies the nature of the operating software included within the download file. For instance, the code can be operating software, boot-up software, or a software revision table. Again, if the software identifier is incorrect, the download is immediately aborted. A build release field 90 identifies the version number for the operating software file. In one embodiment, the version number is incorporated into the file name, which is included in the build release field 90. The cable modem 20 compares the file name received with the file name stored in memory (representing the last used version of the operating software). If the file names match, it is not necessary for the cable modem 20 to download the operating software.

A vendor identification field 92 identifies the vendor and cable modem model number (or other identification number) on which the operating software will run. When the cable modem is manufactured, the vendor stores a vendor identification number in the cable modem memory, so that it cannot later be changed or corrupted. If the vendor identification number set forth in the vendor identification field 92 does not match the number stored in the cable modem, then the software download aborts. Spare bytes in the header 70 are indicated by a reference character 94. It is understood by those skilled in the art that the number of bytes occupied in the spare field 94 can vary depending upon the length of the header and the length of other header fields.

In one embodiment, the operating software 68 is downloaded from a TFTP server and therefore follows the TFTP protocols. As is know by those skilled in the art, in accord with the TFTP protocol, the code is downloaded is 512 bite packets. Assuming typical field lengths, the field 70, 72, 74 and 76 will generally comprise less than 512 bytes. Therefore the analysis in accordance with the present invention occurs after 512 bytes have been downloaded. This analysis proceeds sequentially from the header field 70 and its constituent fields, through the various other fields comprising the operating software file 68. In another embodiment, when the software is not downloaded from a TFTP server, the cable modem can independently check each field when the download for that field had been concluded. In another embodiment, the header field 70 is downloaded and then the various fields associated therewith are evaluated before continuing with the download of the field 72, 74, 76

8

and 78. In any case, when analysis of one of the fields indicates a software version mismatch or another problem, the download aborts and an error is generated as discussed above.

Finally, the hardware class field 96 identifies the cable modem classes (i.e. models or groups of models) on which the operating software code in the field 78 will run. Since there are 256 bits in the hardware class field 96, 256 different hardware modem classes can be supported. The hardware identifier (which is typically eight bits long) stored in the cable modem memory is used as an index into the 256 bitmap of the hardware class field 96. If the indexed bit is set, then that hardware is supported by the operating software in the field 78. The advantage of utilizing a bitmap derives from the fact that any given operating software code can support more than one hardware class. All supported hardware classes are incorporated into the bitmap.

Having now reached the point in the initialization process where all of the fields 82 through 96 have been checked and no problems identified, the operating software code field 78 of Figure 3 is downloaded. The validity of the operating software code is determined using the checksum value in the field 76. Then the checksum value in the field 84 is utilized to analyze the entire downloaded file, including all fields set forth in Figure 3. If no problems are detected by these two checksum evaluations, the cable modem 30 reboots using the downloaded operating software. In another embodiment either one or both of the checksums 76 and 84 can also be used to correct one or more detected errors.

Figure 5 illustrates the process of evaluating each of the various parameters set forth in Figures 3 and 4. The Figure 5 flow chart begins at a step 110 when the cable modem 20 powers up. Note that the analysis process illustrated in Figure 5 does not necessarily have to occur in the order shown in Figure 5. At a step 112, the target identification field 86 is compared to the device type as stored in the device, in particular a cable modem 20. Recall that software for different device types is available on the cable system 10. If the result of the step 112 is negative, the software download process is aborted at a step 114.

If the target identification comparison process results in an affirmative response from the decision step 112, processing moves to a decision step 116 where

9

the software identification field 88 is evaluated to determine the type of software in the field 78 (see Figure 3). If the result of the decision step 116 indicates that the software is not appropriate or is not the correct software type, processing moves from the decision step 116 to the step 114 where the download process is aborted. An affirmative response from the decision step 116 moves the process to a decision step 118. At this step, the software version information (the field 90 of Figure 4) is evaluated. If the version is the same as the software version previously used by the cable modem 20, then it is unnecessary to download the software. Under those circumstances, processing moves from the decision step 118 to the abort step 114.

If the decision step 118 indicates that this is a new software version, processing moves to a decision step 120 where the hardware class field 96 is evaluated. The objective of the decision step 120 is to ensure that the modem hardware class is supported by the software. If that class is not supported, the process moves from the decision step 120 to the abort step 114. If the modem hardware class is supported, the software is downloaded at a step 122. Next, error identification and correction is performed using a checksum value downloaded with the software. This process of error identification and correction can encompass only the downloaded software or all of the various fields associated with the header 70, using the checksum values in the checksum fields 76 and 94. The error identification and correction process is illustrated at a step 124. If no errors were discovered or all the errors were correctable, the result from a decision step 126 is affirmative and the cable modem is rebooted using the downloaded software (see a step 128). If the result of the decision step 126 indicates that the downloaded software includes uncorrectable errors, then, as shown at a step 130, the modem utilizes the previous version of the software, instead of the version downloaded at the step 122.

While the invention has been described with reference to a preferred embodiment, it will be understood by those skilled in the art that various changes may be made and equivalent elements may be substituted for elements thereof without departing from the scope of the present invention. In addition, modifications may be made to adapt the teachings of the present invention to a particular situation without departing from the essential scope thereof. Therefore, it is intended that the invention not be limited to the particular embodiment disclosed as the best mode contemplated

for carrying out this invention, but that the invention will include all embodiments falling within the scope of the appended claims.

WHAT IS CLAIMED IS:

1.   A method for determining whether software intended for download to a device is correct and valid software for that device, wherein the device includes a hardware identifier, wherein the hardware identifier further includes a device type and a device class, and wherein a header preceding the software includes a target identification field and a hardware class field, said method comprising:

(a)   determining if the device type is consistent with the target identification field;

(b)   determining if the device type is consistent with the hardware class field; and

(c)   downloading the software if the results of steps (a) and (b) are affirmative..

2.   The method of claim 1 wherein the hardware identifier is permanently stored within the device at the time of device manufacture.

3.   The method of claim 1 wherein the device type identifies the device as a cable modem.

4.   The method of claim 1 wherein only the header is downloaded to the device, after which the software is downloaded if the results of steps (a) and (b) are affirmative.

5.   The method of claim 1 wherein the hardware class field is a multiple-bit bitmap, wherein the device class is a binary value, and wherein the multiple-bit bitmap is compared with the binary value for determining if the hardware class field matches the device class.

6.   The method of claim 5 wherein the hardware class field bitmap includes 256 bits and wherein the device class includes a binary value having eight bits, and wherein the step (b) further comprises using the eight-bit hardware as an index into the 256 bits to determine if an index bit is set.

7.   The method of claim 1 wherein the header further includes a build release field identifying the software version, and wherein the method further comprises a step (b1) determining whether the software version as set forth in the build release field is different from the software version last used by the device and executing the download step (c) if the result of step (b1) is affirmative.

12

8.   The method of claim 1 wherein the software is device operating software.

9.   The method of claim 1, wherein the header further includes a checksum for determining the software validity, the method further comprising:

5   (d)   performing error identification on the software using the checksum.

10.   The method of claim 9 further comprising:

(e)   correcting errors in the software using the checksum.

11.   The method of claim 1 wherein the step (a) is executed immediately after the target identification field is downloaded.

10   12.   The method of claim 1 wherein the step (b) is executed immediately after the hardware class field is downloaded.

13.   The method of claim 1, wherein the header includes a checksum, further comprising:

(d)   performing error analysis on the header and the software using the
15   checksum, after the download of step (c).

14.   The method of claim 13 wherein the error analysis includes error identification.

15.   The method of claim 13 wherein the error analysis includes error identification and correction.

20   16.   The method of claim 1 wherein the header includes a software type identifier indicating the software function, the method further comprising:

(d)   at the device, determining if the indicated software type is applicable to the device; and

(e)   downloading the software in response to step (d).

25   17.   The method of claim 1 wherein the header includes a software version value, the method further comprising:

(d)   at the device, determining if the software version is required by the device;

(e)   downloading the software in response to step (d)

30   18.   A method of determining whether to download software intended for a device and further for determining whether the software is correct software for that device and further for determining whether the software is valid, wherein the device

13

includes a hardware identifier comprising a device type, a vendor identification, and a hardware class, and wherein code portion preceding the software comprises a first checksum, a second checksum, a target identification field, a software identification field, a software release field, a vendor identification field and a hardware class field, said method comprising:

(a)   determining if the target identification field is consistent with the device type;

(b)   determining if the software identification field indicates a software type appropriate for the device;

(c)   determining if the software version is appropriate for the device;

(d)   determining if the vendor identification field is consistent with the vendor identification of the device;

(e)   determining whether the hardware class field is consistent with the hardware class of the device;

(f)   downloading the software in response to the steps (a) through (e);

(g)   performing error identification on the downloaded software using the first checksum;

(h)   performing error identification on the downloaded software plus the preceding code portion using the second checksum.

19.   An article of manufacture comprising:

a computer-readable medium having computer-readable program code embodied therein for determining whether software intended for download to a device is the correct software for that device, wherein the device includes a hardware identifier comprising a device type and a device class, wherein a header preceding the software comprises a target identification field and a hardware class field, said article of manufacture comprising:
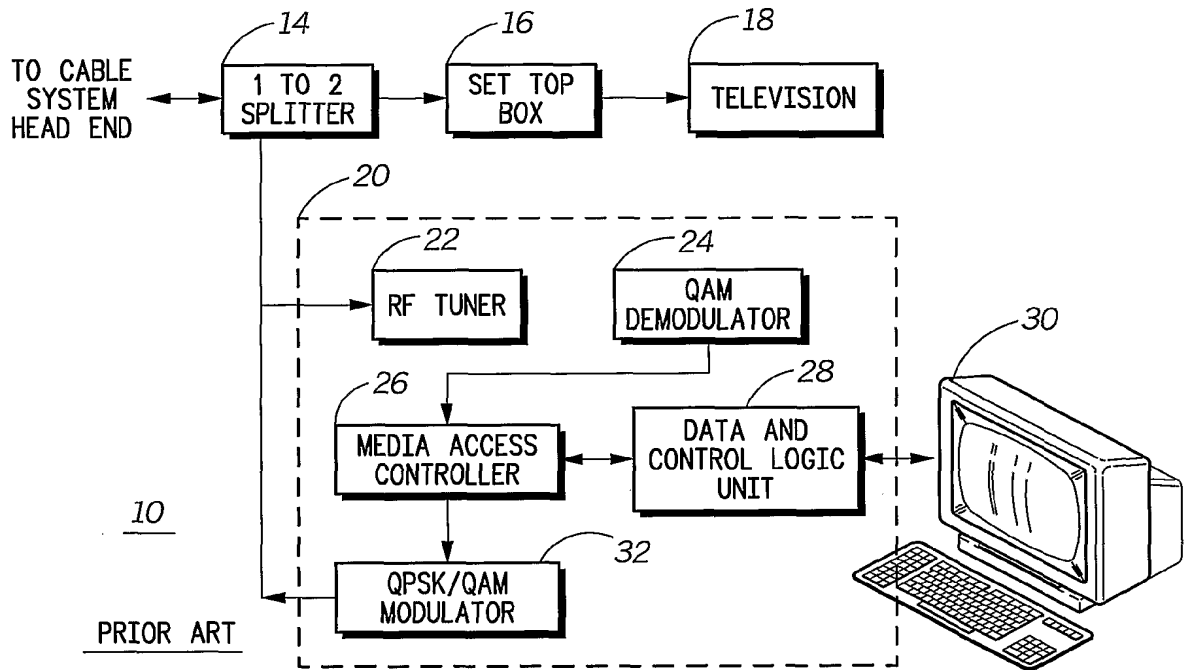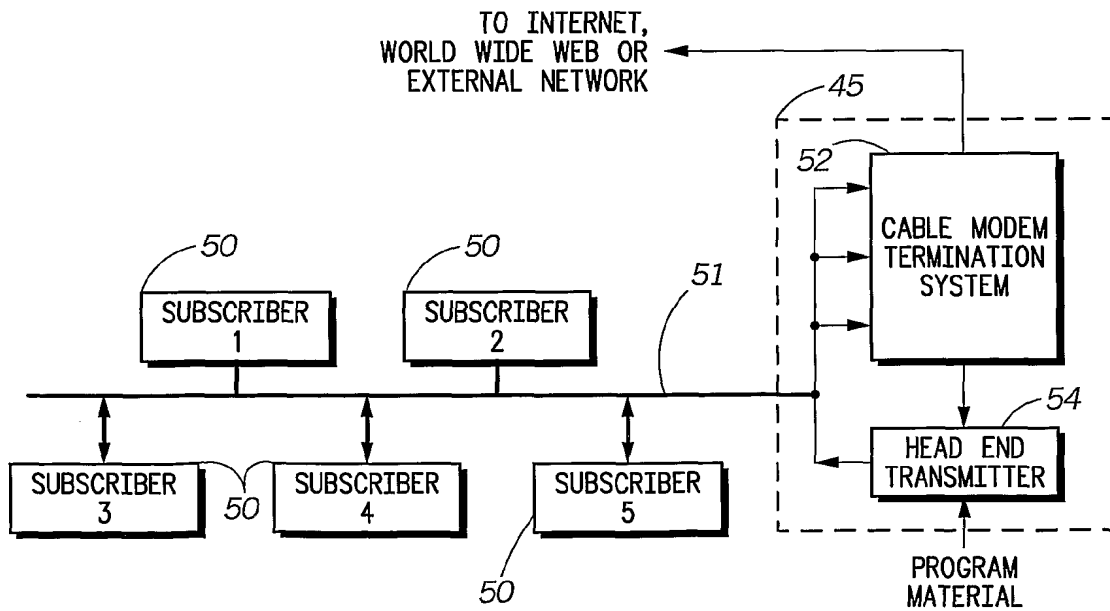
(a)   computer-readable program code configured to cause a computer to determine if the hardware class field matches the device class portion of the hardware identifier; and

(b)   computer-readable program code configured to cause a computer to download the software.

14

20.   An apparatus for determining whether software intended for download to a device is the correct software for that device, wherein that device includes a hardware identifier comprising a device type and a device class and wherein a header preceding the software comprises a target identification field and a hardware class field, said apparatus comprising:

(a)   a first comparator for determining if the target identification field matches the device type portion of the hardware identifier;

(b)   a second comparator for determining if the hardware class field matches the device class portion of the hardware identifier; and

(c)   a controller for downloading the software in response to said first and second comparators when the target identification field matches the device type portion of the hardware identifier and the hardware class field matches the device class portion of the hardware.

1/3

TO CABLE
SYSTEM ◄─────► [14] 1 TO 2 SPLITTER ───► [16] SET TOP BOX ───► [18] TELEVISION
HEAD END

[20]

[22] RF TUNER   [24] QAM DEMODULATOR

[30]

[26] MEDIA ACCESS CONTROLLER ◄───► [28] DATA AND CONTROL LOGIC UNIT

10

[32] QPSK/QAM MODULATOR

PRIOR ART

FIG. 1

TO INTERNET,
WORLD WIDE WEB OR ◄──────
EXTERNAL NETWORK

[45]

[52]

[50] SUBSCRIBER 1   [50] SUBSCRIBER 2   [51]   CABLE MODEM TERMINATION SYSTEM

[50] SUBSCRIBER 3   [50]   SUBSCRIBER 4   SUBSCRIBER 5   [54] HEAD END TRANSMITTER

50

PROGRAM
MATERIAL

PRIOR ART

FIG. 2

| HEADER |
| DECOMPRESSION CODE |
| LENGTH |
| CHECKSUM |
| OPERATING SOFTWARE CODE |

70 → HEADER
72 → DECOMPRESSION CODE
74 → LENGTH
76 → CHECKSUM
78 → OPERATING SOFTWARE CODE

68

FIG. 3

| LENGTH |
| CHECKSUM |
| TARGET ID |
| SW ID |
| BUILD RELEASE |
| VENDOR ID |
| SPARE |
| HARDWARE CLASS |

82 → LENGTH
84 → CHECKSUM
86 → TARGET ID
88 → SW ID
90 → BUILD RELEASE
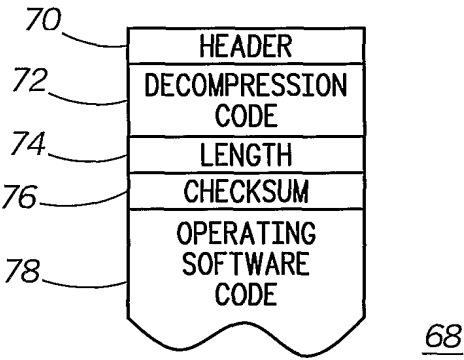92 → VENDOR ID
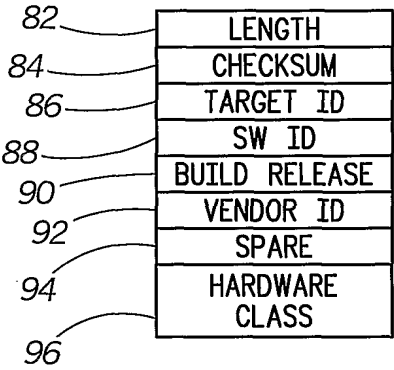94 → SPARE
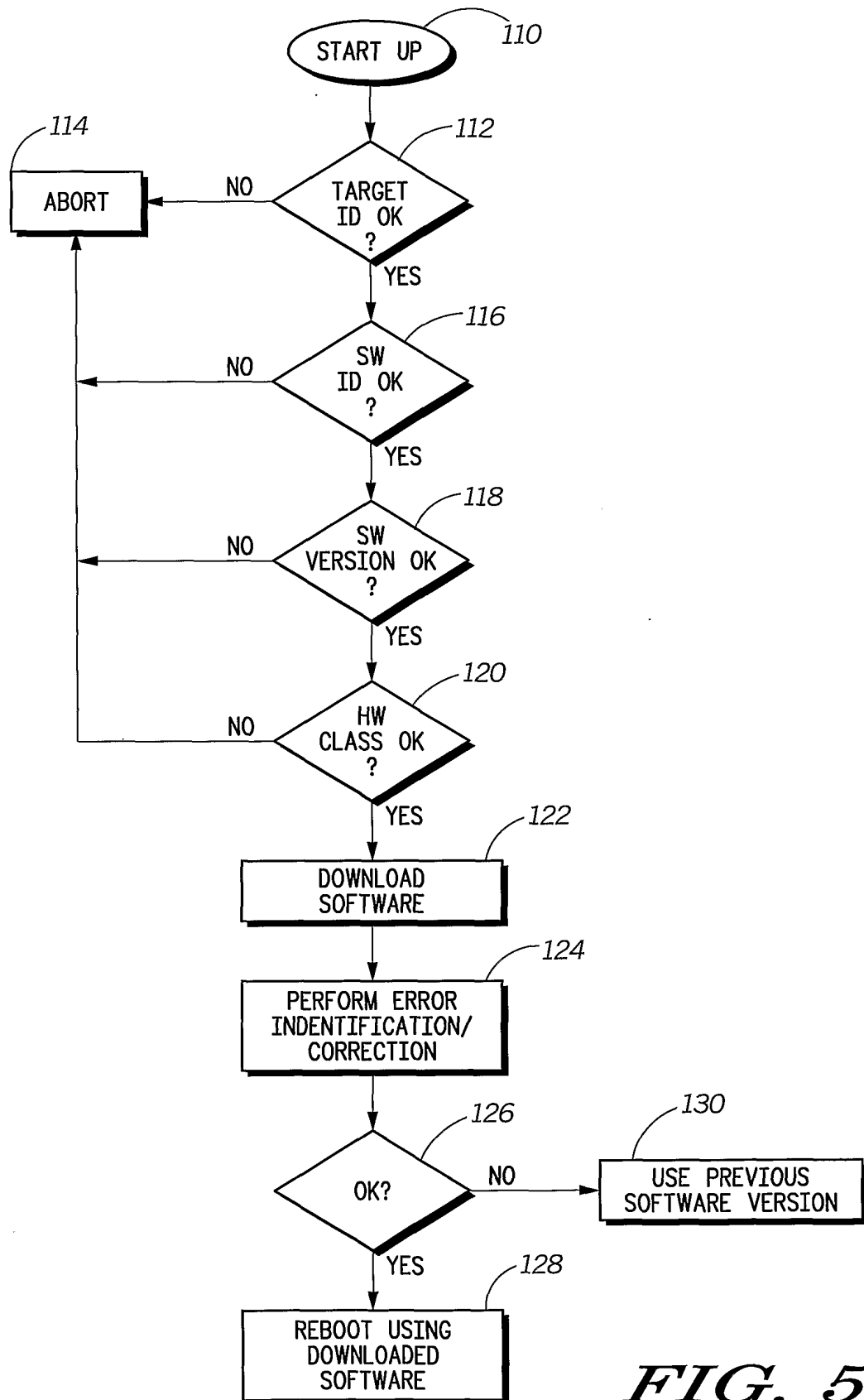96 → HARDWARE CLASS

FIG. 4

3/3

FIG. 5

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US01/50124

A.   CLASSIFICATION OF SUBJECT MATTER
   IPC(7)   :   H04B 7/08
   US CL   :   713/202.00
According to International Patent Classification (IPC) or to both national classification and IPC

B.   FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
   U.S. : 713/160, 190-194,202 ; 370/475,537,486,498

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
Please See Continuation Sheet

C.   DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X, P	US 20010001611 A1 (Yuzawa) 24 May 2001 (24.05.2001) , pg. 2, paragraphs 21-25, pg. 4, paragraphs 53-60, pg. 5, paragraphs 61- 67, 71-72, see also claims 1-2,Fig. 1-2. pg. 6 through 7, paragraphs 81-89, and 98.	1-20
Y	EP 828386 A2 (Ezaki et al.) 11 March 1998 (11.03.1998), column 4, lines 45-57 through column 5, lines 1-11, column 5, lines 20-30, lines 49-53, fig. 10B, 10 C and 10D, claims 1, 2, 3, 4, 5 11, 23 and 24.	1-20
Y	US 5,416,840 A (Cane et al.) 16 May 1995 (16.05.1995), abstract, column 2, lines 19-65, column 3, lines 64-68 throgh column 4, lines 1-7, lines 57-68, see also Fig. 4, claims 1, 2 and 6.	1-20
Y	US 5, 745,568 A (O' Connor et al.) 28 April 1998 (28.04.1998), entire documnet.	1-20
A, E	US 20020010864 A1 (Safa) 24 Januaray 2002 (24.01.2002), entire diocument.	1-20

☐   Further documents are listed in the continuation of Box C.   ☐   See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance		
"E"	earlier application or patent published on or after the international filing date	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed	"&"	document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
18 March 2002 (18.03.2002)	18 APR 2002

Name and mailing address of the ISA/US
    Commissioner of Patents and Trademarks
    Box PCT
    Washington, D.C. 20231
Facsimile No. (703)305-3230
	Authorized officer
	Gail O Hayes James R. Matthews
	
	Telephone No. (703) 305-4274

Form PCT/ISA/210 (second sheet) (July 1998)

Continuation of B. FIELDS SEARCHED Item 3:
WEST, Dialog; Search terms: target Identification; Set top Identification; Software Identification; Device type and Model; packet header and hardware identification.